UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/692,946	10/24/2003	Konstantin Levit-Gurevich	30320/17232	2114

4743 7590 02/09/2007

MARSHALL, GERSTEIN & BORUN LLP
233 S. WACKER DRIVE, SUITE 6300
SEARS TOWER
CHICAGO, IL 60606

EXAMINER	
ALHIJA, SAIF A	
ART UNIT	**PAPER NUMBER**
2128	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/09/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/692,946	LEVIT-GUREVICH ET AL.	
	Examiner	Art Unit	
	Saif A. Alhija	2128	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 November 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,2 and 4-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2 and 4-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *24 October 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-2, and 4-22 have been presented for examination.

Claim 3 has been cancelled.

Response to Arguments

2. Applicant's arguments filed 14 November 2006 have been fully considered but they are not persuasive.

i) Following Applicants arguments and amendment the 112 rejections have been withdrawn.

ii) Examiner notes Applicants acknowledgment of the Double Patenting rejection provided in the previous office action.

iii) Applicant argues the following:

> *Devine et al.*, however, does not teach the recited subject and cannot because each VMM of *Devine et al.* is only capable of monitoring a single VM. Thus, *Devine et al.* does not teach creating "a plurality of virtual machines in a virtual environment, the virtual environment being a direct execution environment," executing "virtual code on at least one of the plurality of virtual machines" or providing "a monitor within the host environment to control entry to and exit from each of the plurality of virtual machines in the direct execution environment." As *Devine et*

> In addition to being in condition for allowance by dependency from claim 1, applicant separately highlights that *Devine et al.* also fails to teach the subject matter of claims 20 and 21. The prior art, for example, does not teach or suggest an apparatus capable of assigning "the virtual code to any one of the plurality of virtual machines for execution" or monitoring "the plurality of virtual machines for a virtualization event." Furthermore, the prior art does not teach or suggest determining "if the virtualization event is an end of quota event for one of the

> ...rality of virtual machines" or "in response to an identification of the end of quota event, [switching] to another one of the plurality of virtual machines."

iv) However, Column 1, Lines 53-64 of the reference states:

> Virtual machine monitors were popular at a time where hardware was scarce and operating systems were primitive. By virtualizing all the resources of the system, multiple independent operating systems could coexist on the same machine. For example, each user could have her own virtual machine running a single-user operating system.
>
> The research in virtual machine monitors also led to the design of processor architectures that were particularly suitable for virtualization. It allowed virtual machine monitors to use a technique known as "direct execution," which simplifies the implementation of the monitor and improves performance. With direct execution, the VMM sets up the

This indicates that the VMM operates over multiple independent operating systems so each user can have their own virtual machine. This anticipates a VMM with multiple VM's as well as a direct execution environment.

Column 11, Lines 34-48 of the reference also states:

> The memory tracing mechanism implemented in the preferred embodiment of the invention uses a combination of the processor's memory management unit (MMU), via page faults, and the ability, using either hardware or software (in particular, the binary-translation sub-system) to execute instructions one-by-one, that is, to single-step the virtual machine. The memory tracing mechanism can be implemented on top of the mechanism that virtualizes the physical address space of the virtual machine. This latter mechanism is present in conventional virtual machine monitors that support multiple virtual machines and can be implemented using known techniques. In the preferred of the invention, it is implemented by having the VMM manage the MMU through an address space separate from the one managed by the VM.

This indicates that the Virtual Machine Monitor supports multiple Virtual Machines. See lines 42-45 above.

Column 25, Lines 7-21 of the reference further states:

> The use of multiple processors allows, for example, the simultaneous execution of multiple virtual machines, or the execution of a virtual machine with multiple virtual processors. The virtualization of each processor is handled separately and independently, including the decision as to whether to use direct execution or binary translation. For each virtual processor, the VMM will then maintain a separate set of global and local shadow descriptor entries.
>
> One minor extension needed when using the invention in a multi-processor environment involves the memory tracing mechanism, which is used, for example, by the VMM for segment tracking and translation cache coherency. What is required is some interaction between processors when traces are installed and when traced accesses occur. This interaction between processors can be implemented using known techniques such as inter-processor interrupts (IPI's).

This section indicates that a VMM can support multiple VM's and also that the VMM utilizes specific memory tracing to support multiple VM's. Applicant argues a part of this section, lines 7-14, however taken as a whole the VMM utilizes a single monitor with respect to at least memory/processor interaction when supporting multiple VM's.

v) Please note the 112 2nd rejections regarding newly submitted claims 21 and 22 and the subsequent claim interpretation and rejections provided below.

Double Patenting

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d

887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4. **Claims 1-19 are provisionally rejected** on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1-26 of copending Application No. 10/395557. Although the conflicting claims are not identical, they are not patentably distinct from each other because the claims in the instant application are directed to a plurality of virtual machines in a direct execution environment and it would have been known to one of ordinary skill in the art at the time of the invention to utilize the invention as directed in the co-pending application in a plurality rather than singular environment.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. **Claims 21 and 22 are rejected** under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

i). Claims 21 and 22 recite the phrase "end of quota event." It is unclear what is meant by an end of quota event. The phrase "end of quota event" is not explicitly defined in the specification. This can be seen in paragraph 32 where quota is first mentioned as merely an example. The only time "end of quota event" is not mentioned in the context of an example is in paragraph 35, which states "block 328 checks if the exit Event was due to the simulated processor end of quota." This does not resolve the indefiniteness of the phrase. Is the end of quota the end of available resources for the processor? Or perhaps the end of use of the processor? Or perhaps the end of code execution by the processor? The phrase therefore renders the claim vague and indefinite.

Appropriate correction is required.

All claims dependent upon a rejected base claim are rejected by virtue of their dependency.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

1-2, 4-22

6. **Claims ~~1-22~~ are rejected** under 35 U.S.C. 102(b) as being clearly anticipated by **Devine et al. "Virtualization System Including a Virtual Machine Monitor for a Computer with a Segmented Architecture", U.S. Patent No. 6,397,242,** hereafter referred to as **Devine.**

Regarding Claim 1:

Devine discloses An article comprising a machine-accessible medium having stored thereon instructions that; when executed by a machine, cause the machine to:

execute a host code in a host environment; **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

create a plurality of virtual machines in a virtual environment, the virtual environment being a direct execution environment; **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

transfer a virtual code from the host environment to the virtual environment; **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

and execute virtual code on at least one of the plurality of virtual machines; and **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

provide, a single monitor within the host environment to control entry to and exit from each of the plurality, of virtual machines in the direct execution environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 2:

Devine discloses The article of claim 1, having further instructions that, when executed by the machine, cause the machine to:

execute the host code in a host operating system environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 4:

Devine discloses The article of claim 3, having further instructions that, when executed by the machine, cause the monitor to: control transfer of virtual code between the host environment and the virtual environment based on a virtualization event attempted by at least one of the virtual machines. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 5:

Devine discloses The article of claim 4, having further instructions that, when executed by the machine, cause the monitor to gain control over the virtualization event from the direct execution environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 6:

Devine discloses The article of claim 5, having further instructions that, when executed by the machine, cause the monitor to return execution to the direct execution environment after a virtualization operation. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 7:

Devine discloses The article of claim 5, having further instructions that, when executed by the machine, cause the monitor to pass control to a platform simulator within the host environment for simulation of the virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 8:

Devine discloses The article of claim 4, having further instructions that, when executed by the machine, cause the monitor to access a list of virtualization events. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 9:

Devine discloses The article of claim 3, having further instructions that, when executed by the machine, cause the monitor to: in response to an exit from the direct execution environment, store state data; and restore the stored state data prior to entry to the direct execution environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 10:

Devine discloses The article of claim 1, wherein the virtual code includes a plurality of virtual codes each executing on a separate one of the plurality of virtual machines. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 11:

Devine discloses A method comprising: accessing simulated instruction codes in a host environment operating on a central processing unit (CPU) implementing Virtual Machine Extensions; launching a plurality of virtual machines in a virtual environment on the CPU, the virtual environment being a direct execution environment; virtualizing a CPU state associated with the simulated instruction codes; executing at least one of the simulated instruction codes on at least one of the plurality of virtual machines; and monitoring, via a single monitor, each of the plurality of virtual machines within the host environment to control entry to and exit from each of the plurality of virtual machines. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 12:

Devine discloses The method of claim 11 further comprising: detecting an occurrence of a virtualization event in any one of the plurality of virtual machines; in response to detecting the virtualization event, exiting the virtual environment; and analyzing the virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 13:

Devine discloses The method of claim 12 further comprising: determining whether the virtualization event is a complex event; and if the virtualization event is not a complex event, virtualizing the simulated instruction code associated with the virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-**

64. Column 11, Lines 34-48. Column 25, Lines 7-21)

Regarding Claim 14:

Devine discloses The method of claim 13 further comprising re-entering the virtual environment after the simulated instruction code associated with the virtualization event is virtualized. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 15:

Devine discloses The method of claim 13 further comprising: if the virtualization event is a complex event, de-virtualizing the CPU state; and simulating the simulated instruction code associated with the virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 16:

Devine discloses The method of claim 12, further comprising: storing the CPU state upon exiting the virtual environment; and restoring the stored CPU state upon re-entering the virtual environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 17:

Devine discloses A system comprising: hardware to generate and control a plurality of virtual machines that each are capable of executing simulated instruction code, wherein the hardware is able to

create an abstraction of a real machine for executing a real operating system on the computer system is not impeded; a direct execution environment to execute the simulated instruction codes and associated data as virtual codes; a plurality of virtual machines formed within the direct execution environment; and a host environment comprising a single monitor for controlling exit from and entry to the plurality of virtual machines formed within the direction execution environment. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 18:

Devine discloses The system of claim 17, wherein the monitor generates the plurality of virtual machines and performs virtualization operations, and wherein the host environment comprises: a platform simulator to perform simulations of virtualization events. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 19:

Devine discloses The system of claim 18, wherein the monitor gains control from the direct execution environment whenever at least one of the plurality of virtual machines attempts to perform a virtualization event, and wherein the monitor switches from one of the plurality of virtual machines to another of the plurality of virtual machines in response to an analysis of the virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 20:

Devine discloses The article of claim 1, having further instructions that, when executed by the machine, cause the monitor to:

assign the virtual code to any one of the plurality of virtual machines for execution; **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)** and

monitor the plurality of virtual machines for a virtualization event. **(Column 5, Lines 12 - Column 6, Line 52. Column 10, Lines 51-59. Column 24, Lines 2-13. Figures 1-2, and 7-8) (Column 1, Lines 53-64. Column 11, Lines 34-48. Column 25, Lines 7-21)**

Regarding Claim 21:

Devine discloses The article of claim 20, having further instructions that, when executed by the machine, cause the monitor to:

determine if the virtualization event is an end of quota event for one of the plurality of virtual machines; and

in response to an identification of the end of quota event, switch to another one of the plurality of virtual machines. **(Claim Interpretation. An end of quota event is interpreted to be a situation where a virtual machine is no longer active either due to lack of resources, lack of code execution, or lack of use and another virtual machine can be "run" in its absence. This can be seen in Devine in at least the MMU of the VMM which deals with memory allocation for the VM's. The MMU prevents incorrect memory mapping and supports handling/switching between multiple VM's with respect to memory allocation/de-allocation to the multiple VM's that require it. Column 11, Line 45 – Column 12, Line 67)**

Regarding Claim 22:

Devine discloses The method of claim 12, further comprising:

determining if the virtualization event is an end of quota event for one of the plurality of virtual machines; and

in response to an identification of the end of quota event, switching to another one of the plurality of virtual machines. **(Claim Interpretation. An end of quota event is interpreted to be a situation where a virtual machine is no longer active either due to lack of resources, lack of code execution, or lack of use and another virtual machine can be "run" in its absence. This can be seen in Devine in at least the MMU of the VMM which deals with memory allocation for the VM's. The MMU prevents incorrect memory mapping and supports handling/switching between multiple VM's with respect to memory allocation/de-allocation to the multiple VM's that require it. Column 11, Line 45 – Column 12, Line 67)**

Conclusion

7. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

8. All Claims are rejected.

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Saif A. Alhija whose telephone number is (571) 272-8635. The examiner can normally be reached on M-F, 11:00-7:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kamini Shah can be reached on (571) 272-22792279. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SAA

January 13, 2007

HUGH JONES Ph.D.
PRIMARY PATENT EXAMINER
TECHNOLOGY CENTER 2100